

UNIVERSIDAD TÉCNICA FEDERICO SANTA MARÍA
DEPARTAMENTO DE INFORMÁTICA
VALPARAISO - CHILE



**“DISEÑO E IMPLEMENTACION DE UNA PLATAFORMA
WEB PARA LA RESOLUCION DE PROBLEMAS CON
MÚLTIPLES CRITERIOS”**

OSCAR MANUEL DUARTE MARIN

**MEMORIA PARA OPTAR AL TÍTULO DE
INGENIERO CIVIL EN INFORMÁTICA**

Profesor Guía: Dr. María Cristina Riff Rojas
Profesor Correferente: Dr. Nicolás Emilio Rojas Morales

OCTUBRE – 2019

DEDICATORIA

“...Y solo él era mi amigo de verdad, Firulais”

AGRADECIMIENTOS

RESUMEN

Resumen— La toma de decisiones puede ser un proceso complejo si se considera en ella la participación de muchas alternativas y objetivos, si a eso se le añade la posibilidad de que múltiples personas opinen, el escenario se complica aún más.

En esta memoria se diseñará e implementará una plataforma web que dará soporte a la toma de decisión con múltiples criterios usando la técnica *Analytic Hierarchy Process* (AHP). Para llevar esto a cabo se realiza una breve reseña del fundamento teórico de la técnica y una revisión del *software* existente. En base a ello se seleccionan requerimientos para la implementación de la plataforma, dando como resultado un software que finalmente se valida, apoyándose en la comparación con un *software* existente y pruebas con usuarios.

Palabras Clave— AHP; plataforma web; decisión con múltiples criterios

ABSTRACT

Abstract— Decision making can be a complex process if the participation of many alternatives and objectives are considered, if is added the possibility that multiple people give their opinion, the scenario becomes even more complicated.

In this memoir, a web platform will be designed and implemented that will support decision making with multiple criteria using the Analytical Hierarchy Process (AHP), To do this, a brief review of the theoretical basis of the technique and a review of the existing software is made. Based on this, requirements for the implementation of the platform are selected, resulting in a software that is finally validated, supported on the comparison with existing software and tests with users.

Keywords— AHP; web platform; *multi-criteria decision problem*

GLOSARIO

Índice de consistencia: función matemática que da cuenta de cuán transitivos (consistentes, coherentes) son las preferencias del tomador de decisiones.

Índice de consistencia geométrico: índice de consistencia planteado en base a los valores recíprocos de la matriz de comparación de a pares.

Índice de determinantes: índice de consistencia basado en la teoría de determinantes de matrices.

Alternativa: curso de acción u opción considerada como parte del proceso de toma de decisiones. Se utiliza indistintamente como curso de acción y opción.

Análisis de sensibilidad: analiza los efectos de cambios que se realizan sobre ciertos elementos de decisión para evaluar cómo y/o cuánto cambia el resultado del modelo.

Atributo: característica de un criterio o subcriterio.

Back-end: En diseño de software, el *back-end* es la parte que procesa la entrada desde el *front-end*.

Comparación de a pares: procedimiento que involucra dos elementos y que emite un juicio. Contrasta dos o más elementos de decisión, utilizando una cierta escala, que permite discriminar cuál es mejor respecto a cierto objetivo. Formalmente, se traduce en la entrada de una matriz.

Consistencia: medida que indica el nivel de relación lógica entre elementos.

Criterio: consideración que se utiliza para preferir una alternativa, usualmente relacionado a atributos. Se usa indistintamente con objetivo, sin embargo, este último término en ocasiones hace referencia al sentido de mejora de los atributos o criterios.

Curso de acción: alternativa u opción considerada como parte del proceso de toma de decisiones. Se utiliza indistintamente como opción y alternativa. Decisión con múltiples criterios problemas en los que intervienen diferentes criterios y que, generalmente, entran en conflicto entre ellos.

Desempeño: también denominado score, es un número que representa la importancia relativa de un elemento de decisión al elegir un curso de acción entre un conjunto de opciones. Generalmente se aplica a alternativas.

Desviación cuadrática: medida del error cometido en el método de priorización.

Elemento de decisión: variable o parámetro considerado en la toma de decisión (meta, criterio, alternativa o atributo).

Escala ordinal: medida categórica que asigna rangos a elementos y que permite ordenarlos comparándolos como “más alto” o “mejor” y “más bajo” o “peor”.

Meta: fin perseguido por el tomador de decisiones. También hace referencia a la combinación de elemento de decisión tal que se considere como óptima, bajo ciertos criterios de aceptación, es decir, la mejor alternativa.

Front-end: En diseño de software, el front-end es la parte del software que interactúa con los usuarios.

Framework: es una estructura conceptual y tecnológica de asistencia definida, normalmente, con artefactos o módulos concretos de *software*, que puede servir de base para la organización y desarrollo de *software*

Java-Applet: En Java, un applet es un programa que puede incrustarse en un documento HTML, es decir, en una página web. Actualmente no son soportados por la mayoría de los navegadores web

Lixiviación: Extracción de materia soluble desde una mezcla, mediante la acción de un disolvente líquido.

Método de la columna normalizada: método de priorización utilizado para estimar el vector de prioridades de una matriz de comparación de a pares. Para esto es necesario el normalizar sus columnas, es decir, que sumen 1. Luego se suma por filas y se divide en la cantidad de columnas, obteniendo el vector de prioridades.

Método de la media geométrica: método de priorización utilizado para estimar el vector de prioridades de una matriz de comparación de a pares, donde cada elemento de el vector

corresponde a la media geométrica de las columnas normalizada por la media geométrica de los elementos de la matriz.

Método del vector propio: método de priorización utilizado para estimar el vector de prioridades de una matriz de comparación de a pares haciendo, uso de la información que entregan sus valores propios.

Método de la media geométrica revisada: método de priorización que construye una matriz de comparación de a pares auxiliar y permite aproximar el vector de prioridades cuando algunas de sus entradas son desconocidas.

Método de priorización: procedimiento utilizado para obtener un vector de prioridades, es decir, resumir información en el proceso de toma de decisión.

Objetivo: dirección de mejora de atributos o criterios, es decir, consideración que se utiliza para seleccionar qué alternativa es mejor. Se usa indistintamente con criterio.

Opción: curso de acción u opción considerada como parte del proceso de toma de decisiones. Se utiliza indistintamente como alternativa y curso de acción.

Open-source: término que hace referencia a software que cumple con todas las normas definidas en la especificación Open Source Definition publicada en el sitio oficial de la institución Open Source Initiative y que se caracteriza, a grandes rasgos, porque su código fuente y otros derechos bajo licencia de código abierto.

Peso: es un número que representa la importancia relativa de un elemento de decisión respecto a otro. Generalmente se aplica a criterios y subcriterios. También se denomina *weight*.

Rank reversal: ocurre cuando los pesos de dos criterios o alternativas cambian de tal manera que producen un cambio en el orden del ranking.

Ranking: orden de preferencia de las alternativas basado en sus desempeños.

Random index: valor estimado promedio del índice de consistencia para matrices aleatorias de tamaño n .

Razón de consistencia: versión escalada de índice de consistencia.

Tomador de decisiones: quien posee la tarea de discriminar entre diferentes alternativas buscando dar respuesta a un cierto problema.

Vector de prioridades: también conocido como vector de pesos, resume el desempeño de algún elemento de decisión.

Violaciones de prioridades: medida del error cometido en el método de priorización.

GLOSARIO DE ACRÓNIMOS

AHP: el Proceso Analítico Jerárquico (en inglés Analytic Hierarchy Process), es una técnica estructurada para tratar decisiones complejas, para ello realiza comparación de a pares.

API: código, ya sea funciones o procedimientos, que permite la comunicación de dos o más software.

GUI: tipo de interfaz que permite al usuario interactuar con software a través de iconos e indicadores gráficos.

SMART: técnica para apoyar la toma de decisiones complejas, basada en un modelo lineal aditivo, es decir, el desempeño global de una alternativa está determinado por cómo se comporta respecto a cada criterio multiplicado por el peso de dicho criterio.

UML: lenguaje gráfico para visualizar, especificar y documentar cada una de las partes que comprende el desarrollo de software.

INDICE DE CONTENIDOS

RESUMEN.....	4
ABSTRACT	4
GLOSARIO DE ACRÓNIMOS	9
INDICE DE FIGURAS.....	12
INDICE DE TABLAS.....	13
INTRODUCCIÓN	14
CAPÍTULO 1: DEFINICIÓN DEL PROBLEMA	15
1.1 Objetivos	17
1.1.1 Objetivo general.	17
1.1.2 Objetivos específicos.	17
CAPÍTULO 2: MARCO CONCEPTUAL	18
2.1 Proceso de decisión.....	18
2.1.1 Formulación.	18
2.1.2 Comparación a pares.	20
2.1.2.1 Escalas de comparación.....	20
2.1.3 Proceso Jerárquico.....	21
2.1.4 Priorización y consistencia	24
2.1.5 Medidas de error	25
2.1.6 Análisis de sensibilidad	25
2.2 Revisión del <i>software</i> AHP	26
2.3 Revisión API AHP	30
CAPÍTULO 3: PROPUESTA DE SOLUCIÓN	31
3.1 Software vs Plataforma web	31
3.2 Selección de lenguajes de programación, librerías, frameworks y tecnologías	31

DISEÑO E IMPLEMENTACION DE UNA PLATAFORMA WEB PARA LA RESOLUCION DE PROBLEMAS CON MULTIPLES CRITERIOS

3.3 Modelo de Datos	35
3.4 Construcción de la interfaz gráfica.....	35
3.4.1 Soporte del proceso	36
3.4.2 Construcción del modelo y aplicación de métodos	36
3.4.3 Aplicación de métodos	37
3.4.4 Análisis de resultados	38
3.4.5 Otros elementos	40
CAPÍTULO 4: VALIDACIÓN DE LA SOLUCIÓN	42
4.1 Pruebas con usuarios	42
4.1.1 Metodología y resultados	44
4.2 Pruebas de respuesta	47
CAPÍTULO 5: CONCLUSIONES	48
5.1 Revisión de objetivos	48
5.2 Trabajo futuro	49
REFERENCIAS	50
ANEXOS.....	52

INDICE DE FIGURAS

Ilustración 1: Jerarquía AHP	22
Ilustración 2: Ejemplo jerarquía AHP.....	24
Ilustración 3: Relaciones entre componentes	34
Ilustración 4: Captura de software, pasos de soporte	36
Ilustración 5: Construcción del modelo en la plataforma	37
Ilustración 6: Formas de ingreso de comparación	37
Ilustración 7: Selección de métodos en la plataforma	38
Ilustración 8: Resultados del problema	38
Ilustración 9 : Análisis probabilístico	39
Ilustración 10: Análisis de sensibilidad	39
Ilustración 11: Rank Reversals	40
Ilustración 12: Documentación de problemas	41
Ilustración 13: Resultados obtenidos en la plataforma	45
Ilustración 14: Resultados pruebas Expert Choice	45
Ilustración 15: Resultados obtenidos en la plataforma	46
Ilustración 16: Resultados pruebas Expert Choice	46

INDICE DE TABLAS

Tabla 1: Escalas de comparación relativa.....	21
Tabla 2: Versiones de Softwares	33
Tabla 3: Modelo de datos	35
Tabla 4: Criterios del modelo AHP.....	43
Tabla 5: Modelo AHP	43

INTRODUCCIÓN

Una decisión se puede considerar como el resultado o respuesta de un proceso que se define como toma de decisiones. Este proceso busca incluir la evaluación de las posibles alternativas, para escoger aquella que satisfaga una meta u objetivo según una serie de parámetros y criterios. Estos componentes son conocidos como elementos de decisión, y su correcta definición y uso repercutirán en el éxito que pueda tener un tomador de decisiones. En este ámbito de toma de decisiones surge *Analytic Hierarchy Process* (AHP) [1], una metodología que da soporte a la toma de decisión con múltiples criterios en donde existen muchas alternativas y, posiblemente, participantes.

Este trabajo es una extensión del realizado en [2], que proporciona una *Application Programming Interface* (API) para resolución de problemas de decisión de múltiples criterios.

Este trabajo se centra en la construcción de una plataforma web que actuará como interfaz gráfica entre el usuario y la API de resolución de problemas AHP. Además, se revisa el estado del arte de software que resuelven problemas de múltiples criterios y cómo ha sido su evolución en estos últimos años. Esto permite definir los requerimientos que se considerará el software a construir. Luego se propondrá una solución a estos requerimientos.

Se analiza el correcto funcionamiento y desempeño de la solución propuesta, incluyendo pruebas y comparaciones con otros softwares a modo de corroborar su correcto funcionamiento. Para finalizar, se presentan las conclusiones que se obtuvieron de la realización del trabajo y se describe trabajo futuro, así como los resultados que se esperan a largo plazo.

CAPÍTULO 1: DEFINICIÓN DEL PROBLEMA

A diario se toman cientos de decisiones, de manera que se puede decir que cada una de nuestras acciones tiene una decisión involucrada. Por lo general, estas están basadas o fundamentadas en opiniones, pero este trabajo se enfoca en las decisiones que podemos cuantificar sus criterios, parámetros y por ende su objetivo, para esto existen áreas dedicadas al estudio de cómo tomar las mejores decisiones que por lo general son denominadas como “Teoría de Decisión” [3]. Para modelar cualquier tipo de problema de decisión son necesarios ciertos pasos: en primer lugar, se necesita definir el problema para identificar cuál es la meta, luego se identifican cuáles son las alternativas u opciones que existen para solucionar el problema, a continuación, se deben valorar las consecuencias de cada alternativa u opción, asignando una medida cuantificable a cada característica, previamente definida. En base a eso se elige la alternativa que más se adecúe a la meta u objetivo impuesto en un principio bajo cierto criterio o criterios.

Uno de los elementos que estudia la teoría de decisión, y busca resolver de la manera óptima, son las decisiones con múltiples criterios (DMC). En este caso resulta confuso o poco claro cuál es la mejor alternativa a el problema, puesto que existen varios criterios. Este estudio se enfoca en este tipo de problemas y en la utilización de la técnica conocida como *Analytic hierarchy process* para su modelamiento y resolución.

Los problemas de decision con múltiples criterios se pueden caracterizar y resolver en los siguientes pasos:

1. Formulación y descripción: determinar actores, criterios a considerar, estructura organizativa, es decir, interdependencias entre participantes, contexto del problema, conjunto inicial de alternativas.
2. Modelado: formular, expresar y/o traducir el problema a un conjunto de elementos de decisión.
3. Incorporación de las preferencias: categorización, se asigna importancia subjetiva a los elementos de decisión considerados en la etapa anterior. En la mayoría de los casos la importancia resulta ser subjetiva, aunque se podrán asignar preferencias de manera más objetiva si los elementos de decisión lo permiten.
4. Priorización: comparación de resultados obtenidos en base a la información recabada hasta este punto.
5. Retroalimentación: se intenta responder y profundizar en el conocimiento del proceso de decisión en lo relacionado al curso de acción u alternativa seguida, sin limitarse a la resolución del problema para valores particulares.

Aun considerando todos estos pasos, para el tomador de decisiones puede resultar confuso emitir juicio y realizar comparaciones, ya que, en algunas ocasiones se podría encontrar con elementos que dependerán de su subjetividad al momento de ser evaluados y comparados. En dichos casos resultará deseable la realización de un análisis de sensibilidad, que entregará información tanto o más valiosa que la resolución del problema, al entregar información sobre cómo varían las decisiones frente a modificaciones que pueda tener

Si bien hoy en día existen diferentes software que resultan de gran ayuda para resolver un DMC, tienen carencias y que serán analizadas con mayor profundidad en los próximos capítulos:

- Las opciones gratuitas poseen limitaciones en cuanto a número de elementos de decisión o funcionalidades.
- Poseen restricciones respecto al número de elementos de decisión que se pueden añadir al modelo, ya sea criterios, subcriterios o alternativas.
- El análisis de sensibilidad es limitado o no se puede realizar.

La base de esta propuesta es el trabajo realizado en [2], que provee una API para la resolución de problemas utilizando la técnica AHP, y además busca mitigar y resolver de mejor manera los problemas y falencias que tienen los *software* existentes. Ese trabajo deja planeado la construcción de una GUI para facilitar su uso, que presente los resultados de manera gráfica, resumidos y ordenados y con posibilidad de ser exportados.

Considerando todo lo anteriormente planteado, el objetivo principal de este trabajo es el diseño e implementación de una plataforma web para dar soporte a la toma de decisiones con múltiples criterios. Para ello es necesario realizar un estudio del software existente para decisión con múltiples criterios, definir los requerimientos de la plataforma y, finalmente, modelar e implementar integrando el trabajo previamente realizado.

1.1 Objetivos

1.1.1 Objetivo general.

Construcción de una plataforma web para la toma de decisiones con múltiples criterios usando la técnica AHP.

1.1.2 Objetivos específicos.

1. Analizar el software existente para DMC
2. Extraer principales ventajas de cada uno de los softwares para DMC
3. Construcción *back-end* de la plataforma
4. Construcción *front-end* de la plataforma
5. Publicación de software en código abierto.
6. Integración con API AHP desarrollada en [2]
7. Visualización gráfica de los resultados entregados por la API
8. Generación de reportes con los resultados

CAPÍTULO 2: MARCO CONCEPTUAL

De acuerdo con el capítulo anterior, este trabajo se centra en problemas DMC y en particular al uso de la técnica AHP, por lo que en esta sección se resume una conceptualización e investigación teórica del método.

2.1 Proceso de decisión.

Se considera que un proceso de decisión se presenta al momento de comparar una o más alternativas, también denominadas cursos de acción, que son escenarios futuros, cuya elección queda a facultad del tomador de decisiones. Este último las coteja bajo ciertos parámetros, conocidos como criterios, y les asigna una importancia, pudiendo ser objetiva (basada en estadísticas o estudios) o subjetiva (mera apreciación personal). La idea es seleccionar aquella acción que más se aproxima a la ideal bajo los parámetros que antes definió el tomador de decisiones.

2.1.1 Formulación.

Para visualizar el problema se presenta el siguiente ejemplo: se desea seleccionar un automóvil para comprar en un futuro próximo, se consideran diferentes marcas y modelos que luego serán evaluados bajo criterios de interés, por ejemplo: seguridad, salud y transporte, inclinándose finalmente por alguna de las opciones que el mismo proponga. Esta cotidiana situación permite introducir terminología:

Meta: finalidad de la toma de decisión, aquello que se busca lograr, conseguir, maximizar su beneficio. Por ejemplo, seleccionar el mejor automóvil.

Alternativa: posible curso de acción o decisión cuya elección trae consigo ciertos beneficios. En el ejemplo sería Nissan Maxima, Mercedes-Benz 290, Volvo 840 u otros.

Tomador de decisiones: persona, una o más, que tienen la labor de discernir acerca de una o más alternativas, pudiendo elegir una menor cantidad de las que disponen.

Criterio: parámetro bajo el cual una alternativa es evaluada. En el caso anterior, es costo inicial, prestigio y economía de combustible entre otros, notar que son comparadas según estos parámetros, pudiendo ser de manera absoluta (con una escala de calificación) o relativa, en cuyo caso se contrastan de a pares, por ejemplo, en cuanto a coste inicial se refiere Nissan Maxima es dos veces mejor que Mercedes-Benz 290. Hay que destacar que estos criterios pueden dividirse en subcriterios, así calidad podría ser evaluado en seguridad, frecuencia de avería entre otros.

A continuación, se presentan los términos y conceptos a utilizar en este trabajo siendo consistente con [4] y [1]:

- El conjunto de criterios $C = \{ c_j \} \forall j \in \{1, \dots, m\}$
- El conjunto de alternativas $X = \{ x_i \} \forall i \in \{1, \dots, n\}$
- El vector de pesos $W = (w_1, w_2, \dots, w_m)^T$ donde w_i es el peso asociado al criterio i , donde se debe cumplir que $\sum_{j=1}^m w_j = 1$
- El desempeño de todas las alternativas $P^{c_j} = (p_1^{c_j}, p_2^{c_j}, \dots, p_n^{c_j})^T$, donde para cada criterio c_j , se tiene el desempeño $p_i^{c_j}$ para la alternativa x_i , estos también denominados *scores* representan la importancia relativa de un elemento de decisión al elegir un curso de acción u alternativa entre un conjunto de opciones. Además, se debe cumplir que $\sum_{i=1}^n p_i^{c_j} = 1 \forall j \in \{1, \dots, m\}$
- El vector de desempeño global $S = (s_1, s_2, \dots, s_n)^T$ donde s_i es el peso asociado a la alternativa i , donde se debe cumplir que $\sum_{i=1}^n s_i = 1$, además:
 - $s_i = \sum_{j=1}^m p_i^{c_j} w_j$
 - $S = \sum_{j=1}^m w_j P^{c_j}$
- El *ranking* es un orden de preferencia de las alternativas basado en sus desempeños. El vector S define a un ranking en donde se sigue que si $s_i > s_j$ la alternativa x_i es preferida sobre x_j , por ejemplo: sea $S = (1/3, 1/6, 1/2)^T$ implica que $x_3 \succ x_1 \succ x_2$, donde $x_i \succ x_j$ significa que x_i es preferida sobre x_j
- Un método de priorización es un procedimiento utilizado para obtener un vector de prioridades, es decir, resumir información en el proceso de toma de decisión.
- Un vector de prioridades, también conocido como vector de pesos, resume el desempeño de algún elemento de decisión. Son vectores de prioridades W , P^{c_j} y S .

2.1.2 Comparación a pares.

Comparación a pares es un procedimiento que involucra dos elementos y que emite un juicio. Contrasta dos o más elementos de decisión utilizando una cierta escala que permite discriminar cuál es mejor respecto a cierto objetivo. Formalmente, se traduce en la entrada de una matriz de comparación A .

$$A = \begin{pmatrix} 1 & w_1/w_2 & \cdots & w_1/w_n \\ w_2/w_1 & 1 & \cdots & w_2/w_n \\ \vdots & \vdots & \ddots & \vdots \\ w_n/w_1 & w_n/w_2 & \cdots & 1 \end{pmatrix}$$

En donde para cada entrada de la matriz $a_{ij} \approx \frac{w_i}{w_j} \forall i, j$ representa la preferencia de la alternativa x_i sobre x_j , de manera análoga se tiene que $a_{ji} = (a_{ij})^{-1}$, es decir si $a_{ij} = 4$, la alternativa x_i es preferida 4 veces sobre x_j

2.1.2.1 Escalas de comparación.

Según el trabajo realizado por Saaty en [1] y por Brunelli en [5], se proponen dos escalas numéricas positivas y discretas, la escala “fundamental” y la escala “balanceada”, respectivamente por cada autor. Ambas se basan en la información que un ser humano puede procesar con respecto a las alternativas, sin embargo, es necesario mencionar que la determinación de la mejor escala para AHP aún es materia de discusión [6].

Escala		Importancia	Explicación
Fundamental	Balanceada		
1	1	Igual (<i>Equal</i>)	Contribuyen de igual manera al objetivo.
3	1,5	Moderada (<i>Moderate</i>)	Se prefiere un elemento sobre el otro.
5	2,33	Mucha (<i>Strong</i>)	Se prefiere un elemento sobre el otro.
7	4	Demasiada (<i>Very Strong</i>)	Se prefiere fuertemente un elemento sobre el otro.
9	9	Extrema (<i>Extreme</i>)	Evidencia favoritismo (casi forzando su consideración).
2;4;6;8	1,22; 1,86; 3; 5,76	Intermedios	Cuando se requiere compromiso.

Tabla 1: Escalas de comparación relativa
Fuente: Elaboración propia

2.1.3 Proceso Jerárquico

La metodología AHP está definida por los siguientes pasos:

1. Definir el problema y su meta.
2. Estructurar el problema de manera jerárquica que permite de una manera eficiente y grafica organizar la información respecto de un problema, descomponerla y analizarla por partes [7]. El elemento superior es la meta, los niveles intermedios son los criterios sobre los que se evaluará las alternativas. Las alternativas estarán en el nivel más bajo.
3. Construir la matriz de comparación de a pares para cada nodo sin hijos, evaluando cada alternativa respecto a dicho elemento de decisión, y obtener su vector de prioridades. Para cada nivel intermedio construir la matriz de comparación de a pares con los elementos del mismo nivel, evaluando su desempeño respecto a su nodo padre.

4. Utilizar las prioridades obtenidas de las comparaciones para calcular el peso las prioridades en el nivel inmediatamente inferior. Esto se repite para cada elemento. Luego, para cada elemento en el nivel inferior, agregue sus valores ponderados y obtenga su prioridad local o global. Continuar este proceso de calcular el peso y agregar hasta que se obtengan las prioridades finales de las alternativas en el nivel más bajo de la jerarquía.
5. AHP recomienda elegir la alternativa con mejor desempeño.

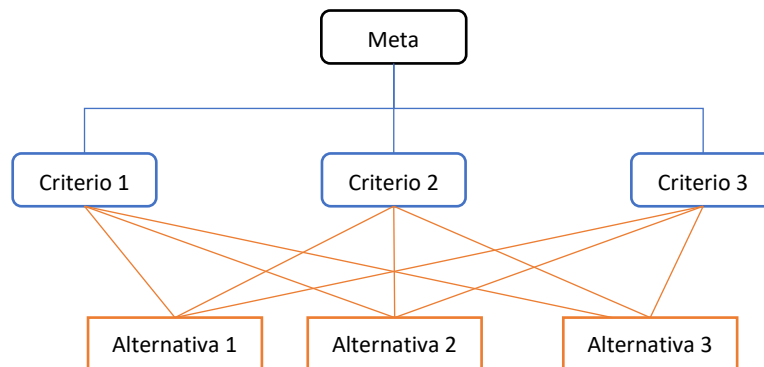


Ilustración 1: Jerarquía AHP
Fuente: Elaboración propia

Para ejemplificar con un problema más complejo, se toma el ejemplo incluido en el software Expert Choice 11.5 en donde se tiene:

- Objetivo: Mejor automóvil.
 - Criterios:
 - Coste inicial
 - Coste de manutención
 - Subcriterios:
 - Coste del Seguro
 - Garantía
 - Economía de combustible
 - Mano de obra y repuestos
 - Prestigio
 - Calidad
 - Subcriterios
 - Seguridad
 - Frecuencia de avería
 - Desempeño
 - Diseño
 - Subcriterios:
 - Interior
 - Exterior
 - Confort de conducción
- Alternativas:
 - Grand Am
 - Nissan Maxima
 - Mercedes-Benz 290
 - Volvo 840
 - Thunder Bird

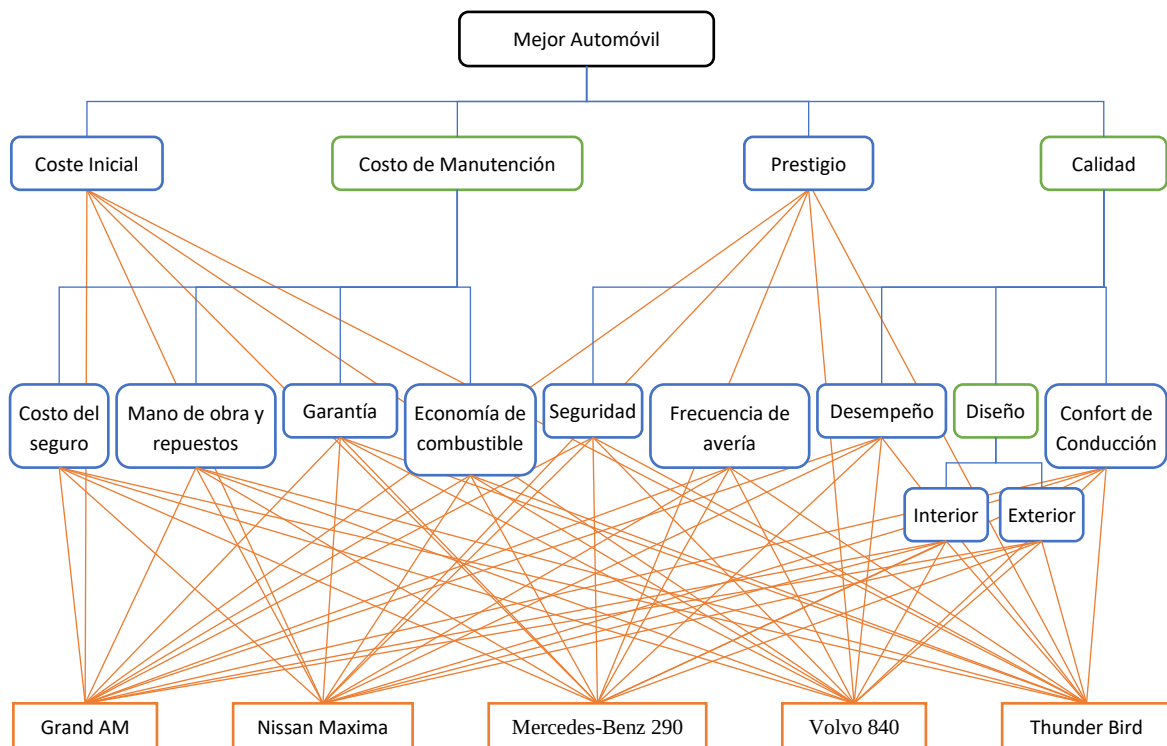


Ilustración 2: Ejemplo jerarquía AHP
Fuente: Elaboración propia.

2.1.4 Priorización y consistencia

En el proceso de decisión es necesario resumir la información para poder realizar una comparación, para esto es necesario un proceso de priorización y así obtener un vector de prioridades, los siguientes son los métodos más conocidos y que están disponibles en la API desarrollada en [2]:

- Método del vector propio
- Método de la media geométrica
- Método de la columna normalizada

Para más detalle sobre estos métodos, se recomienda consultar [2].

Para un tomador de decisiones es necesario conocer que tan transitivos o consistentes son las preferencias que ha incorporado en el modelo, para esto se toma una función llamada índice de consistencia, los siguientes son los métodos más conocidos y que están disponibles en la API desarrollada en [2]:

- Índice de consistencia
- Razón de consistencia
- Índice de determinantes
- Índice de consistencia geométrico

2.1.5 Medidas de error

Para definir si los resultados son buenos y confiables, al calcular los vectores de desempeño para las alternativas, se utiliza una medida de error que permita tener una cuantificación de este. Si el error es muy grande, la alternativa con mejor desempeño puede no ser la indicada para ser elegida.

2.1.6 Análisis de sensibilidad

En ocasiones las comparaciones son realizadas con cierto grado de incertidumbre o son subjetivas y un posible cambio en el tomador de decisiones puede afectar la decisión tomada. Para esto se presentan 2 de los existentes métodos para el análisis de sensibilidad:

1. Análisis numérico incremental

En este método se modifica el peso de un elemento de decisión y se recalcula el *ranking*. Resulta interesante saber cuánto debe cambiar un elemento de decisión para que una alternativa tenga el mismo valor de desempeño que otra, ya que de desde este punto cambiara la elección que sea tomada, a eso se le conoce como *Rank reversal*

2. Simulaciones probabilísticas

Estos métodos reemplazan los valores de la matriz de comparación de a pares con valores de distribuciones probabilísticas y en base a esos cambios se vuelve a calcular el *ranking* general.

2.2 Revisión del *software* AHP

Complementado el trabajo realizado en [2] sobre el análisis de *software* conocidos y utilizados para la resolución de problemas DMC, se tiene lo siguiente:

1. ChemDecide

Software comercial diseñado para selección de rutas, almacenamiento de productos químicos, elección de equipos y decisiones de abastecimiento. Desarrollado para el sistema operativo Microsoft®Windows®.

2. Criterium Decision Plus 3.0

Software comercial desarrollado por Infoharvest, netamente enfocado en el uso gráfico de la herramienta, realizando conexiones entre nodos que representan los elementos de decisión. La versión para estudiantes está desarrollada para sistemas operativos obsoletos (Windows 95®, Windows 98®, Win2000®y Windows XP®).

3. Expert Choice

Actualmente este *software* de pago solo da soporte a una plataforma web que nos provee dos productos: Expert Choice Comparion y Expert Choice Riskion, orientadas a diferentes zonas del mercado. Anteriormente existía una versión para Microsoft Windows a la que se dejó de dar soporte y sus principales ventajas y beneficios son tratados en [8] y [9]. En donde se destaca la incorporación de una interfaz grafica para el usuario intuitiva con calculo automático de prioridades e inconsistencias y varias formas de realizar un análisis de sensibilidad

4. HIPRE 3+

Software gratuito para la resolución de problemas DMC utilizando la técnica AHP y *Simple Multiattribute Rating Technique* (SMART) [10], la última versión fue liberada en abril de 1995 y actualmente se encuentra sin soporte para los sistemas operativos actuales.

5. Web HIPRE

Es un Java-applet gratuito desarrollado en 1998 por la Universidad Politécnica de Helsinki, corresponde a la versión web de HIPRE 3+. Su última actualización fue liberada en septiembre de 2003. Posee análisis de sensibilidad, pero no permite la interacción con el usuario.

6. Logical Decisions

Software de pago para la plataforma Microsoft Windows que posee una versión de prueba de 30 días y de un año para estudiantes. Proporciona una variedad de pantallas de resultados y funcionalidades. El nivel de conocimiento requerido por el usuario es elevado.

7. MakeltRational

Software que cuenta con versiones para Microsoft Windows y una plataforma web, destaca por ser bastante sencillo, enfocado en la parte visual y funcional, herramientas y características muy precisas, incluyendo un sistema de reportes que destaca su capacidad de exportar datos a Microsoft Excel

Este software ya no se encuentra disponible, sus versiones gratuitas no proveían un análisis de sensibilidad [11] y limitaban el número de criterios y alternativas .

8. TransparentChoice

Previamente conocido como MakeltRational, es un software de pago que sólo cuenta con una versión web). Posee una versión gratuita altamente limitada, permitiendo solamente 3 alternativas y 5 criterios para la evaluación además de no contar con ninguna herramienta de análisis de sensibilidad a diferencia de sus versiones de pago que ahora si incluye.

9. PriEsT

Software gratis desarrollado por Siraj et al. [12] , implementado en Java, se dejó de trabajar en él desde 2016. Cuenta con comparación de a pares, múltiples métodos de priorización e índices de consistencia, así como análisis de sensibilidad

10. PUrE2

Es un *software* gratis. Desarrollado para el sistema operativo Microsoft®Windows®y con el objetivo de apoyar las decisiones para la gestión sostenible de la contaminación y energía en edificios, esto hace que sea poco flexible

Mediante el análisis de estos software, es posible detectar algunas falencias que serán de gran utilidad para el correcto desarrollo de este trabajo. Se consideran además otras características no funcionales que parecen no ser tan evidentes, pero serán de gran

repercusión a la hora de tomar una decisión. A continuación, se resume en los siguientes puntos.

1. Limitación de software gratuitos.

Si bien la mayoría de los softwares de pago proporcionan una versión gratuita, es altamente limitada en cantidad de elementos de decisión que se pueden incorporar al modelo y en el análisis de sensibilidad. Es necesario destacar que el análisis de sensibilidad también se encuentra ausente en algunas versiones de pago. Además, se necesita que un *software* sea de uso general y no para la resolución de problemas que están dentro de un contexto bastante específico.

2. Desactualización y no mantención de software.

El avance de los sistemas operativo y en general de los frameworks y lenguajes de programación requiere que los software que son desarrollados estén en constante mantención y actualización.

Quizás el caso más emblemático de este punto sea el de HIPRE en sus 2 versiones, web y 3+, ambas con un gran potencial, que satisfacían en su casi totalidad las necesidades de un usuario para la resolución de problemas DMC. Ambas no han sido mantenidas por lo cual se ha vuelto obsoletas y ni siquiera es posible su instalación en sistemas modernos.

3. Migración a la web

Gran parte de los software de paga han migrado a sus propias plataformas web, lo que proporciona una gran cantidad de beneficios al usuario. No es necesario instalar un programa en un dispositivo con características específicas y/o sistema operativo en cambio se puede ingresar a la plataforma de cualquier parte con un dispositivo que tenga acceso a internet y un navegador.

4. Elementos visuales en software *open-source*

Ningún software sigue alguna normativa de diseño, lo que hace que no sean atractivos para el usuario.

En el trabajo realizado por Jyri Mustajoki y Mika Marttunen [13] se evalúan 24 software de resolución de problemas DMC a través de una estructura definida en este mismo trabajo que separa las características que debería tener un software en cada paso del proceso de decisión. A continuación se presentan los elementos de este trabajo que son aplicables a la técnica de resolución AHP.

1. Soporte del proceso

El uso de un software requiere cierto conocimiento por parte del usuario, que debería ser reducido al máximo. Además, el software debe guiar al usuario en todo el proceso, como es el caso del software V.I.S.A Decisions, que provee un ayudante que guía al usuario en los pasos a seguir para poder modelar y obtener el desempeño de las alternativas

2. Construcción del modelo

En general la construcción del modelo es muy similar entre los software analizados, donde se provee una estructura jerárquica para los criterios. En la mayoría de los casos basado en un árbol, donde se pueden agregar los elementos uno a uno y definir diferentes niveles. En cuanto a la comparación a pares generalmente se presenta una estructura matricial para la selección de las alternativas a comparar y una forma visual para cambiar los valores de la comparación.

3. Aplicación de métodos

El software debe de ser apto para problemas de cualquier contexto, además de proveer un sistema para cuantificar los errores y la inconsistencia del modelo. En el caso particular de AHP, el autor de esta evaluación busca que la comparación a pares sea ingresada por medio de una matriz y que estos valores sean ingresados de alguna manera gráfica, por ejemplo, con el uso de un *slider* entre los dos criterios o alternativas que están siendo comparados.

4. Análisis de resultados

Lo mínimo esperado es que los desempeños de las alternativas sean mostrados de manera gráfica. Idealmente debe contar con un análisis de sensibilidad para ver cómo varían los desempeños de las alternativas. La máxima evaluación de este punto requiere que el análisis de sensibilidad sea probabilístico.

5. Otros elementos

Es deseable que el usuario pueda importar y exportar datos. En el caso de los resultados, es deseable que puedan ser exportados en algún formato basado en tablas Excel. Otro punto deseable es que los software cuenten con una versión web.

2.3 Revisión API AHP

En [2], se diseñó e implementó una API desarrollada en Java para la resolución de problemas DMC utilizando la técnica AHP con las siguientes opciones de métodos:

- Priorización:
 1. Método del vector propio
 2. Método de la media geométrica
 3. Método de la columna normalizada

- Consistencia:
 1. Índice de consistencia
 2. Razón de consistencia
 3. Índice de determinantes
 4. Índice de consistencia geométrico

- Error:
 1. Desviación cuadrática
 2. Violaciones de prioridades

- Análisis de sensibilidad
 1. Análisis numérico incremental
 2. Simulaciones probabilísticas

Es importante mencionar que existe una restricción para el cálculo de consistencia con el método razón de consistencia, limitando la dimensión de la matriz de comparación de criterios o alternativas a un máximo de 15×15 .

En cuanto a su uso, para la resolución del problema ejemplificado en Ilustración 2, son necesarias más de 250 líneas de código para su definición, sin contar las necesarias para extraer los desempeños de las alternativas y el análisis de sensibilidad.

CAPÍTULO 3: PROPUESTA DE SOLUCIÓN

En el capítulo anterior se analizó los principales software de resolución de DMC con la técnica AHP, en conjunto con el trabajo de Jyri Mustajoki y Mika Marttunen [13] se encontraron los elementos que son deseables en un software para este tipo de problemas. A continuación, se diseñará y comentará sobre la construcción de un software que resuelve estos problemas. Se usará el término módulo para referirse a una parte de la plataforma que se comunica con el resto, pero puede funcionar de manera independiente.

3.1 Software vs Plataforma web

Como se vio en el capítulo anterior la mayoría de los softwares que aún continúan con soporte activo han migrado a plataformas web que permiten tanto al desarrollador como al usuario una serie de ventajas. Desde el punto de vista del desarrollador, el uso de una plataforma permitirá su integración con distintos módulos que el desee incorporar a esta, puesto que no requerirá de un completo conocimiento de lo que se ha desarrollado. De esta manera, se intentará construir un *framework* de manera que sea modificable y permita la integración con otros módulos de la manera más sencilla

3.2 Selección de lenguajes de programación, librerías, frameworks y tecnologías

Una restricción de uso de lenguajes para esta plataforma es que se debe usar Java para el uso de la API desarrollada por Quinteros en [2], sin embargo no es el único lenguaje que se usará, ya que como se mencionó anteriormente se intentara construir una plataforma que permita la Inter-operatividad entre diferentes módulos . Todos los lenguajes, framework y tecnologías nombradas a continuación son gratuitas y *open-source*. Por ello, se definen los siguientes módulos que integrarán la plataforma en su versión inicial:

1. Base de datos

Para esto se ha seleccionado mongoDB que es un sistema de bases de datos NoSQL, guardando los datos en estructuras similares a JSON. Se elige por su fácil uso, y dado que el modelo de datos para guardar problemas DMC, no requiere el uso de una base de datos relacional. También, esta base de datos, será la encargada de guardar la documentación para dichos problemas.

2. AHP Server

Dado que la API AHP se encuentra desarrollada en Java, no se ha podido escoger otro lenguaje. Para el desarrollo de este módulo se construyó un servidor en spring-boot que provee un paradigma de desarrollo llamado *convention over configuration*

el cual intenta reducir el número de decisiones que el desarrollador tiene que tomar para el desarrollo de la aplicación, sin quitarle flexibilidad.

Este módulo es el encargado de resolver los problemas que han sido guardados en la base de datos, utilizando la AHP API, para finalmente guardar en la misma los desempeños de las alternativas, así como los análisis de sensibilidad

3. Back-end

Para éste módulo el lenguaje elegido ha sido JavaScript. En particular, con el uso de node.js que es un *runtime environment*, el cual permite la ejecución de los programas desarrollados en JavaScript además de proporcionar un desarrollo rápido al ser de fácil aprendizaje. Además, el *framework* Express proporciona una arquitectura mínima y flexible con un conjunto sólido de características para aplicaciones web. Cabe destacar que node.js es mantenido bajo el soporte de la fundación Linux como además PayPal, Joylent, Microsoft, Walmart.

Este módulo es el encargado de gestionar todo el tráfico de datos en la plataforma.

4. Front-end

Al igual que el en caso del back-end, el lenguaje escogido es JavaScript con el uso de node.js. En este caso se utilizará la librería *React* diseñada para la creación de interfaces de usuario con el objetivo de facilitar el desarrollo de aplicaciones de una sola página. La librería *React* es mantenida por Facebook y la comunidad de software libre

El diseño gráfico de la plataforma se realizará siguiendo la normativa de diseño *material design* que está enfocada en la visualización del sistema operativo Android, además de la web y en cualquier otra plataforma. Intenta mantener un diseño más limpio, en el que predominan animaciones y transiciones de respuesta, el relleno y los efectos de profundidad tales como la iluminación y las sombras. Desarrollada por Google en 2014 y mantenida constantemente a través de los años. La integración de *material desing* con react se realizará mediante el uso de la librería material-ui

5. Nginx

Nginx es un servidor web que también puede ser usado como proxy-reverso y balanceador de carga. De acuerdo con el estudio de Netcraft [14], Nginx es el segundo servidor web más usado en dominios activos (21,71%) .

6. Docker

Docker es un proyecto de código abierto que automatiza el despliegue de aplicaciones dentro de contenedores de software, proporcionando una capa adicional de abstracción y automatización de virtualización de aplicaciones en

múltiples sistemas operativos. De acuerdo con la firma analista de la industria 451 Research [15], "Docker es una herramienta que puede empaquetar una aplicación y sus dependencias en un contenedor virtual que se puede ejecutar en cualquier servidor Linux. Esto ayuda a permitir la flexibilidad y portabilidad en donde la aplicación se puede ejecutar, ya sea en las instalaciones físicas, la nube pública, nube privada, etc".

Para este trabajo cada módulo será empaquetado en un Docker container diferente, permitiendo la Inter-operatividad y la posible incorporación de nuevos containers a la plataforma. En conjunto con Docker será necesario el uso de Docker-compose que permite el despliegue de múltiples containers simultáneamente

La utilización de Docker en este trabajo es de gran importancia ya que se encarga de desplegar la plataforma en cualquier servidor que tenga Docker-compose instalado automatizando la necesidad de la instalación de múltiples librerías y servicios.

7. Autenticación y control de usuarios

Para mantener el control de sesiones de usuario, se han desarrollado dos estrategias: la primera, incorpora un control local utilizando la librería *Passport* y la segunda demostrando la inter-operatividad de la plataforma se incluye la autenticación con la utilización de una cuenta de Google utilizando su arquitectura OAuth2

A continuación, se listan las versiones específicas para cada uno de los softwares mencionados:

Software	Versión
Spring-boot	2.0.5 Release
Nginx	1.13
Node.js	9.4.0
Express	4.14.1
React	15.6.0
Material-ui	3.3.0
Docker	18.09
Docker-compose	1.23.1

Tabla 2: Versiones de Softwares
Fuente: elaboración propia

Siendo relevantes para el despliegue sólo las dos últimas que serán las versiones que deban ser instaladas, la propia plataforma se encargará de la correcta instalación de las demás librerías y softwares.

A continuación, se presenta cómo se relacionan los distintos módulos.

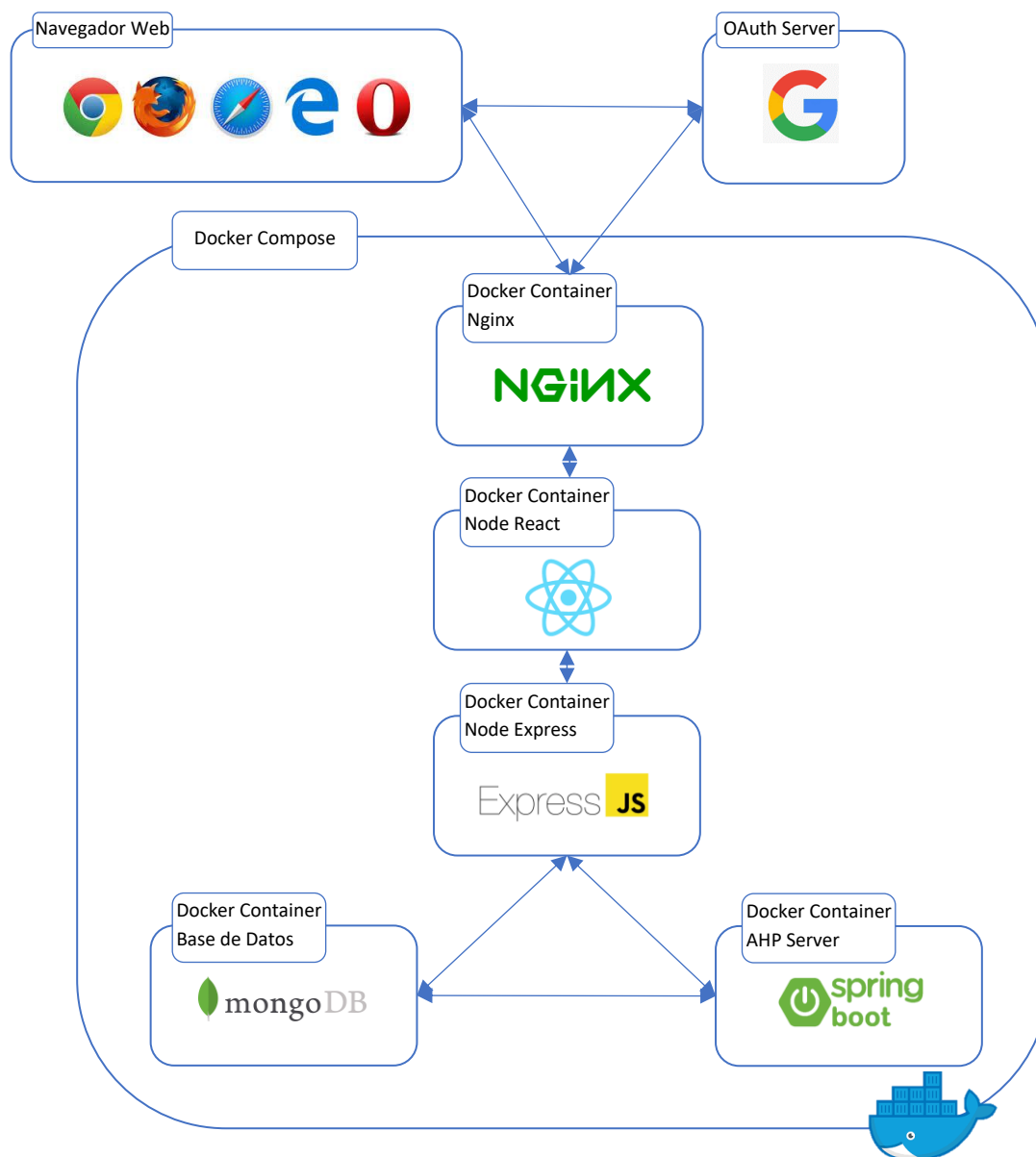


Ilustración 3: Relaciones entre componentes
Fuente: Elaboración propia

3.3 Modelo de Datos

Aprovechando el soporte de modelo de datos tipo objeto, se definen previamente algunos objetos para una mejor visualización y comprensión del modelo de datos.

El objeto denominado *criteria* en las tablas *Ahp_problem*, *Result* y *Sensitivities*, contiene los siguientes campos:

- Nombre
- Subcriterios: es un arreglo de objetos del tipo *criteria* en caso de que el problema contenga subcriterios y así mantener la representación jerárquica.
- Datos: corresponde a los datos relativos a el criterio. Como este objeto se utiliza en varias tablas, tiene significados diferentes, Para *Ahp_problem* será la matriz de comparación del criterio, para *Result* será un arreglo con los desempeños locales de las alternativas y para *Sensitivities* una matriz con los *Rank reversal*.

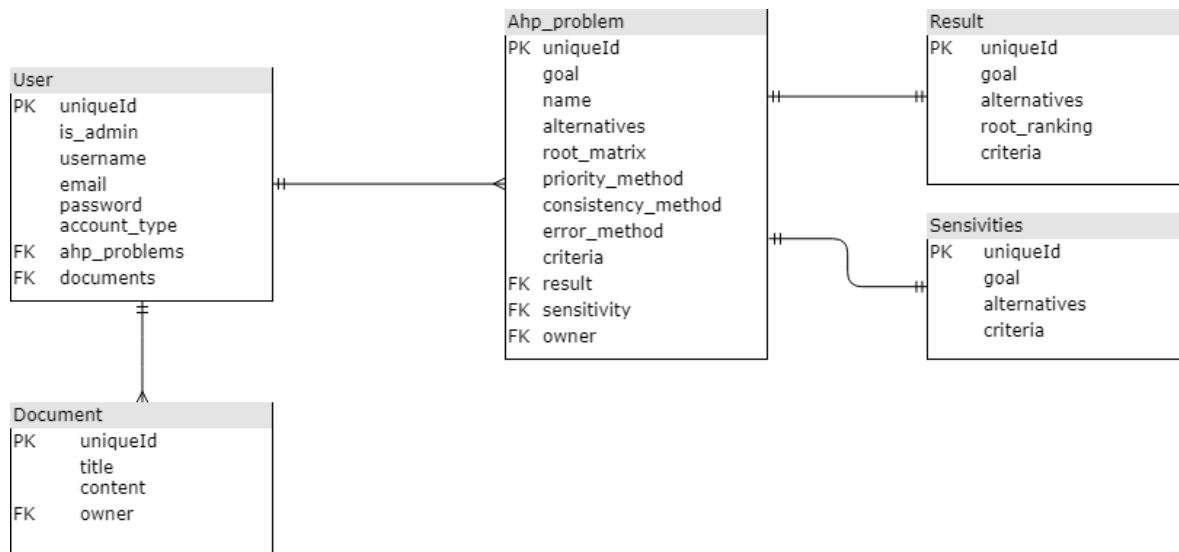


Tabla 3: Modelo de datos

Fuente: elaboración propia

3.4 Construcción de la interfaz gráfica

Siguiendo la evaluación del software realizada en [13], se diseñará el software considerando los puntos que el trabajo considera, como la poder representar la jerarquía de los criterios de manera grafica usando un modelo de árbol, la representación de la matriz de comparación y distintos tipos de sliders para realizar el proceso de comparación.

3.4.1 Soporte del proceso

Durante el proceso de modelar y resolver un problema DMC es necesario minimizar el conocimiento que un usuario deba tener en el proceso. Debido a esto, se ha intentado mantener el diseño para el ingreso del modelo similar a otros programas, con un árbol que modela los criterios y una matriz para el ingreso de las preferencias.

Para mantener guiado al usuario durante el proceso se han creado ciertos pasos en donde el usuario navegará. Estos son:

- Datos del problema
- Parámetros de resolución
- Resultados
- Análisis probabilístico
- Análisis de sensibilidad
- Rank Reversals

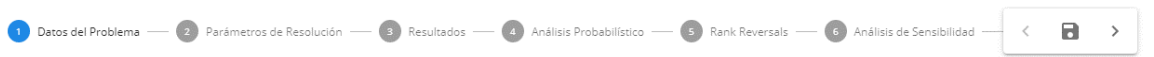


Ilustración 4: Captura de software, pasos de soporte
Fuente: Captura de la plataforma

3.4.2 Construcción del modelo

Para reducir al máximo el conocimiento que deba tener el usuario con respecto al software se han emulado los diseños que mantienen otros software que resuelven problemas DMC con la técnica AHP. Además, se integran ayudas dentro de la misma vista para guiar al usuario novato que no conozca el software.

En la parte izquierda se encuentran los criterios, en donde mediante una estructura jerárquica de árbol de ingresan el objetivo, los criterios y subcriterios. En la parte central, una vez seleccionado un criterio, se muestra la matriz de preferencias en conjunto con los elementos que permiten el ingreso de un valor de preferencia a la matriz. Finalmente, en la parte derecha se encuentra un área para agregar modificar y eliminar las alternativas.

DISEÑO E IMPLEMENTACION DE UNA PLATAFORMA WEB PARA LA RESOLUCION DE PROBLEMAS CON MULTIPLES CRITERIOS

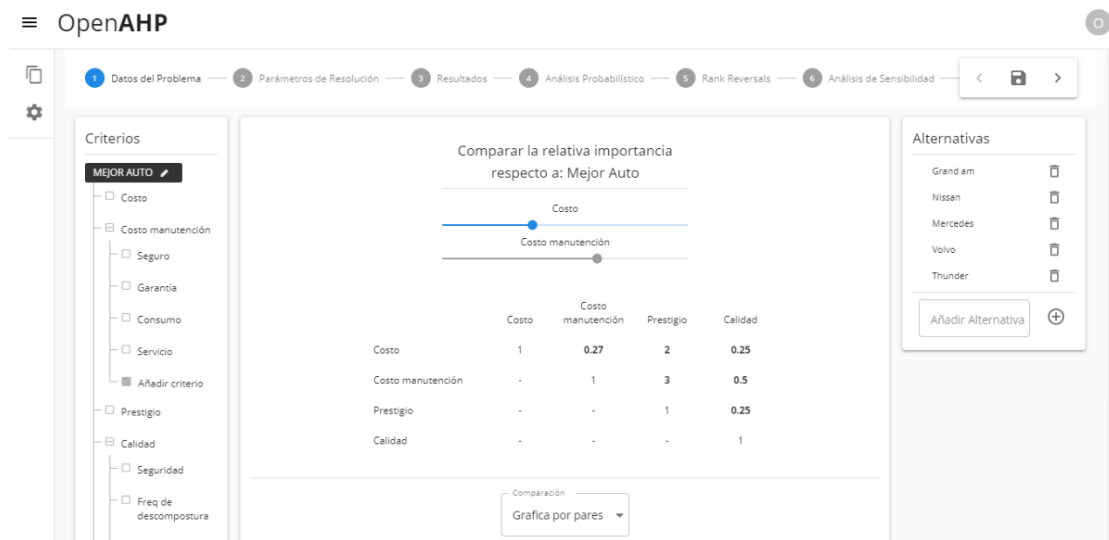


Ilustración 5: Construcción del modelo en la plataforma
Fuente: elaboración propia

3.4.3 Aplicación de métodos

Para el ingreso de preferencias o importancias relativas, respecto a alternativas o criterios respectivamente, se proveen 3 diferentes métodos: el primero basado en la tabla 1 de comparación verbal, y además, 2 métodos por pares de selección, uno gráfico continuo permitiendo al usuario asignar el valor que él estime conveniente, mientras el otro con valores de 1 a 10 con paso de 0.5

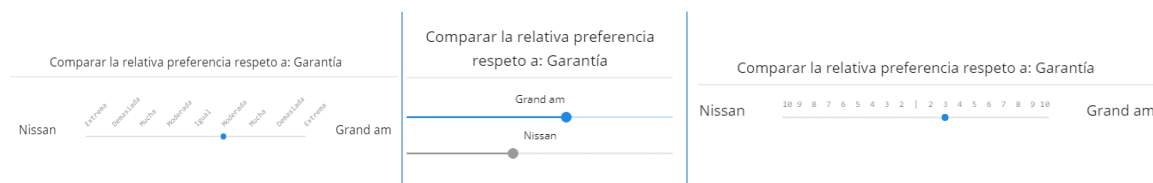


Ilustración 6: Formas de ingreso de comparación
Verbal - Grafica por pares - Por pares
Fuente: Captura de la plataforma

Una vez concluido la incorporación de los datos en el modelo se procede a seleccionar los métodos para el cálculo de prioridades, consistencia y error. Además de los parámetros para el análisis probabilístico

DISEÑO E IMPLEMENTACION DE UNA PLATAFORMA WEB PARA LA RESOLUCION DE PROBLEMAS CON MULTIPLES CRITERIOS



Ilustración 7: Selección de métodos en la plataforma
Fuente: Captura de la plataforma

3.4.4 Análisis de resultados

En este punto se integra una vista que muestra gráficamente los desempeños de las alternativas, tanto globalmente como a nivel local por criterio, seleccionando en el árbol dispuesto a la izquierda de la vista que además muestra el peso obtenido por cada uno. Además, se indican en la parte inferior izquierda el nivel de error e inconsistencia que se tiene para el elemento seleccionado.

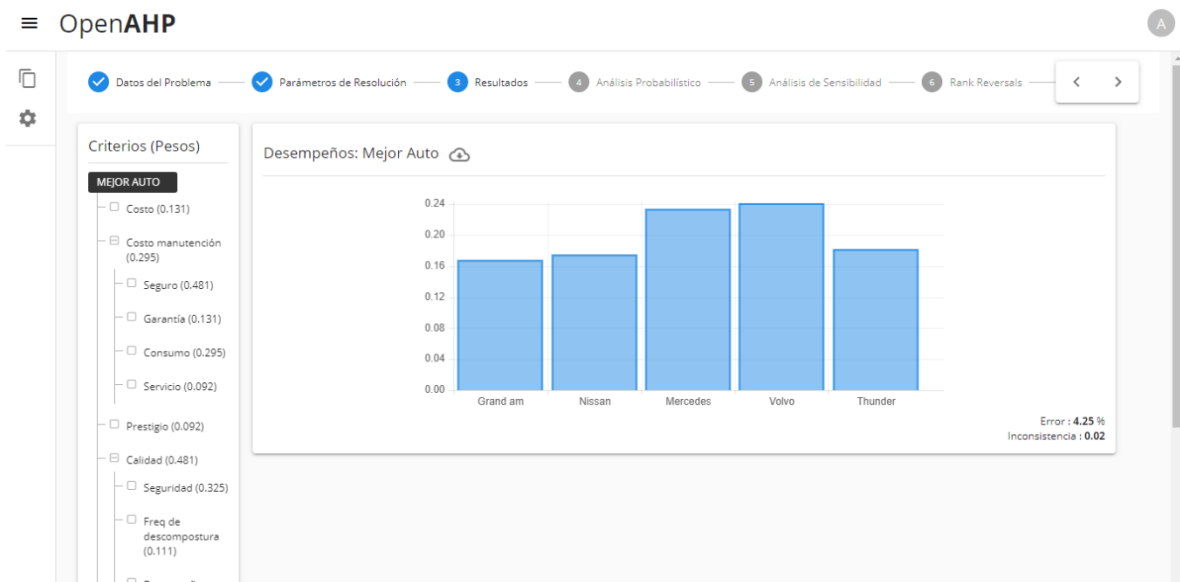


Ilustración 8: Resultados del problema
Fuente: Captura de la plataforma

En el siguiente punto se muestra el resultado de un análisis probabilístico que modifica los valores del peso de los criterios según los parámetros establecidos para mostrar como varían el ranking de las alternativas.

DISEÑO E IMPLEMENTACION DE UNA PLATAFORMA WEB PARA LA RESOLUCION DE PROBLEMAS CON MULTIPLES CRITERIOS

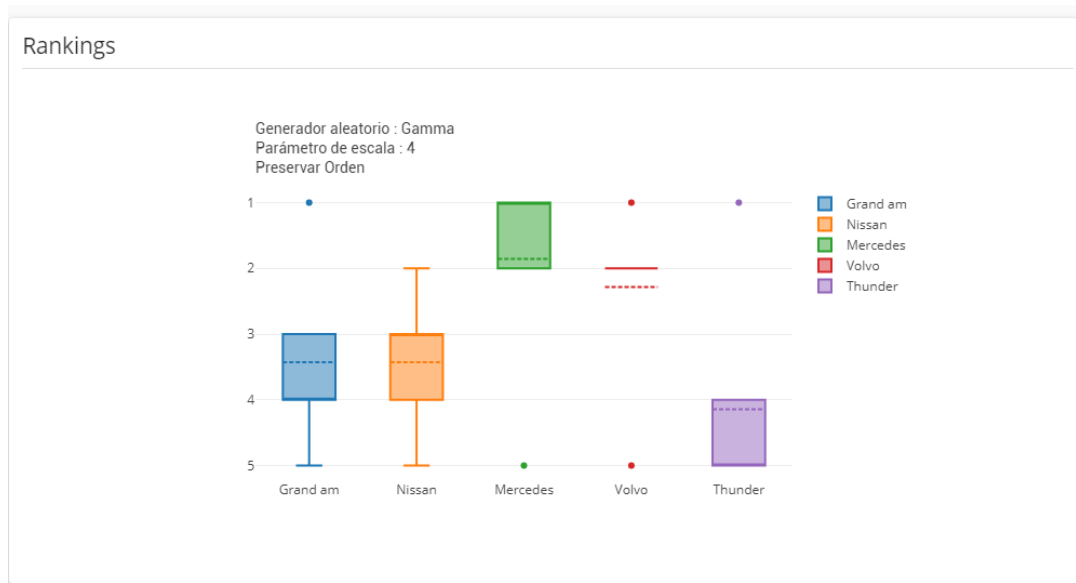


Ilustración 9 : Análisis probabilístico
Fuente: Captura de la plataforma

En el siguiente punto se puede realizar un análisis numérico incremental del problema variando los pesos de los criterios, se presenta de manera que el usuario modifique un peso manualmente y los otros correspondientes al nivel jerárquico se modificaran automáticamente para mantener las restricciones del método.



Ilustración 10: Análisis de sensibilidad
Fuente: Captura de la plataforma

El siguiente punto de esta sección muestra los *Rank reversals*, en donde primeramente se muestran los pesos obtenidos para cada criterio, para luego entregar el peso necesario que debiese tener un criterio seleccionado para que se produzca un *Rank reversal*. Estos son

DISEÑO E IMPLEMENTACION DE UNA PLATAFORMA WEB PARA LA RESOLUCION DE PROBLEMAS CON MULTIPLES CRITERIOS

entregados en una representación matricial en donde el índice ij de la matriz contiene el valor para producir un *Rank reversal* entre las alternativas i y j .

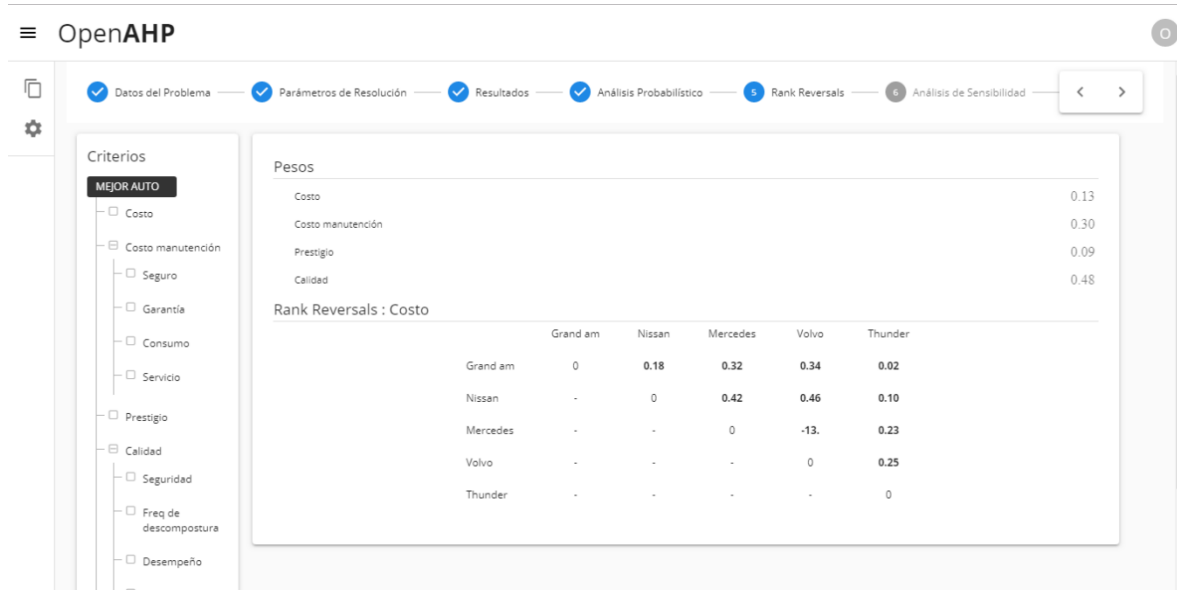


Ilustración 11: Rank Reversals
Fuente: Captura de la plataforma

3.4.5 Otros elementos

Como es recomendado en el estudio de Mustajoki [13], se incluye la exportación de resultados, permitiendo al usuario descargar un archivo con los desempeños de las alternativas. Además, se ha incluido en la plataforma un módulo que permite documentar los problemas, es decir, el autor del problema puede dejar explicitado en texto su modelo u observaciones asociadas a él.

Mejor Auto

- Explicacion del modelo

Lorem Ipsum dolor et sit amen.....

Ilustración 12: Documentación de problemas
Fuente: Captura de la plataforma

CAPÍTULO 4: VALIDACIÓN DE LA SOLUCIÓN

En el capítulo anterior se diseñó una plataforma basándose en un análisis propio como en el análisis realizado en [13]. A continuación, se pondrá a prueba la plataforma construida para corroborar que todas sus funcionalidades respondan a lo esperado. Esto se realizará de la siguiente manera:

1. Se les solicito a 2 personas que no están familiarizados con la plataforma, los problemas DMC o la técnica AHP, resolver el problema disponible en [16] así como el ejemplo puesto en el capítulo. Lo que se busca es ver la facilidad en el uso del software y comprobar los puntos propuestos en [13], es decir cuánto tiempo le toma al usuario acostumbrarse y transcribir un problema a la plataforma y contrastar los resultados obtenidos por este.
2. Se comprobarán que los tiempos de carga y respuesta de la plataforma completa no sean excesivos.
3. Se compararán los resultados entregados por la plataforma con los entregados por el software Expert Choice

4.1 Pruebas con usuarios

El problema expuesto en [16] aborda la elección de una técnica de minería para la extracción de materiales en una mina de níquel, en donde se les proporcionó a los usuarios los siguientes elementos:

- Objetivo: Mejor método de lixiviación
- Alternativas:
 - Lixiviación en pilas
 - Lixiviación con alta presión de acido
 - Lixiviación atmosférica

DISEÑO E IMPLEMENTACION DE UNA PLATAFORMA WEB PARA LA RESOLUCION DE PROBLEMAS CON MULTIPLES CRITERIOS

- Criterios

Criterio	Subcriterio	Descripción
Parámetros de lixiviación	Consumo de ácido	Cantidad de ácido consumido
	Temperatura	El efecto de temperatura en la extracción
	Tiempo de lixiviación	La duración de la lixiviación
	Tamaño de las partículas	La fracción de tamaño de partícula.
	presión	El efecto de presión en la extracción
extracción	Níquel	La extracción de níquel
	Cobalto	La extracción de cobalto
	Hierro	La extracción de hierro
Económico	Inversión inicial	Los costos de inversión de las operaciones tienden a ser significativos y puede ser un factor determinante en la selección de un método de lixiviación adecuado.

Tabla 4: Criterios del modelo AHP

Fuente: Traducción desde [16]

- Comparación de pares: son suministrados la comparación para cada elemento directamente en [16]

A continuación, se presenta el modelo AHP

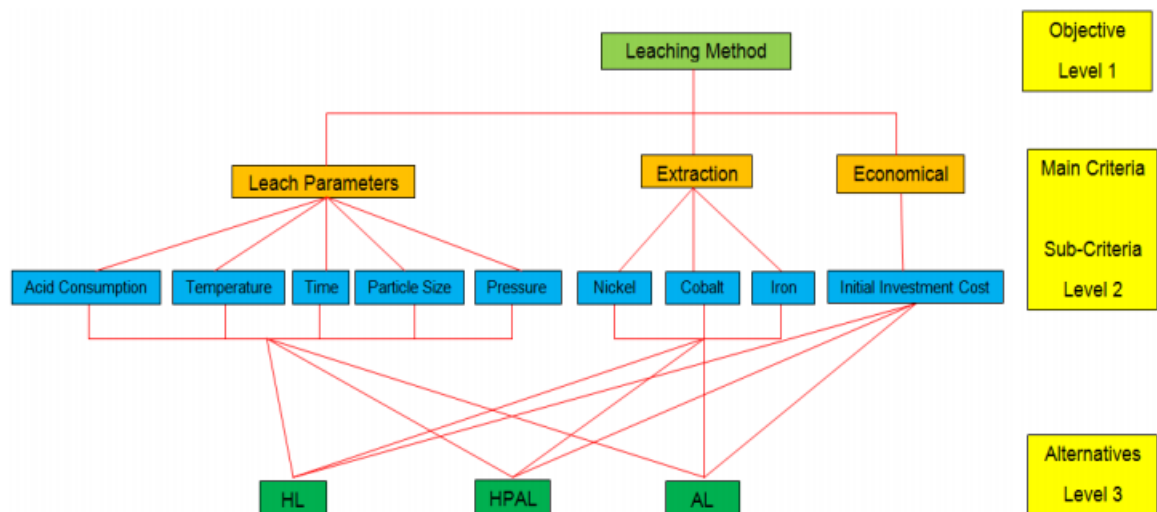


Tabla 5: Modelo AHP

Fuente: [16]

4.1.1 Metodología y resultados

Para el primer problema se les solicitó a los usuarios que resolvieran en la plataforma el problema encontrado en [16], al ser más sencillo en cuanto a número de elementos de decisión, luego se les solicito resolver el ejemplo que se encuentra en el capítulo 2 que tiene mayor número de elementos, adicionalmente se les solicita que resuelvan uno de los problemas a su elección en el software Expert Choice. Se nota lo siguiente:

1. Los usuarios son capaces de resolver por ellos mismos los problemas, no necesitan ayuda externa durante el proceso.
2. El tiempo en que completaron los ejercicios fueron en promedio de 15 minutos para el problema encontrado en [16] y de 20 minutos para el problema propuesto en el capítulo 2. Además, para resolver en Expert Choice un problema al usuario le toman aproximadamente 3 minutos más que en la plataforma
3. La evaluación de las alternativas son similares hasta el tercer decimal comparado con Expert Choice lo que valida los resultados
4. La integración de todo el editor del modelo en una vista resulta más cómodo para el usuario. La guía durante el proceso ayudó a los usuarios a no perder tiempo en buscar opciones dentro del programa.

A continuación, se presentan los resultados obtenidos en el software Expert Choice y en la plataforma desarrollada:

DISEÑO E IMPLEMENTACION DE UNA PLATAFORMA WEB PARA LA RESOLUCION DE PROBLEMAS CON MULTIPLES CRITERIOS

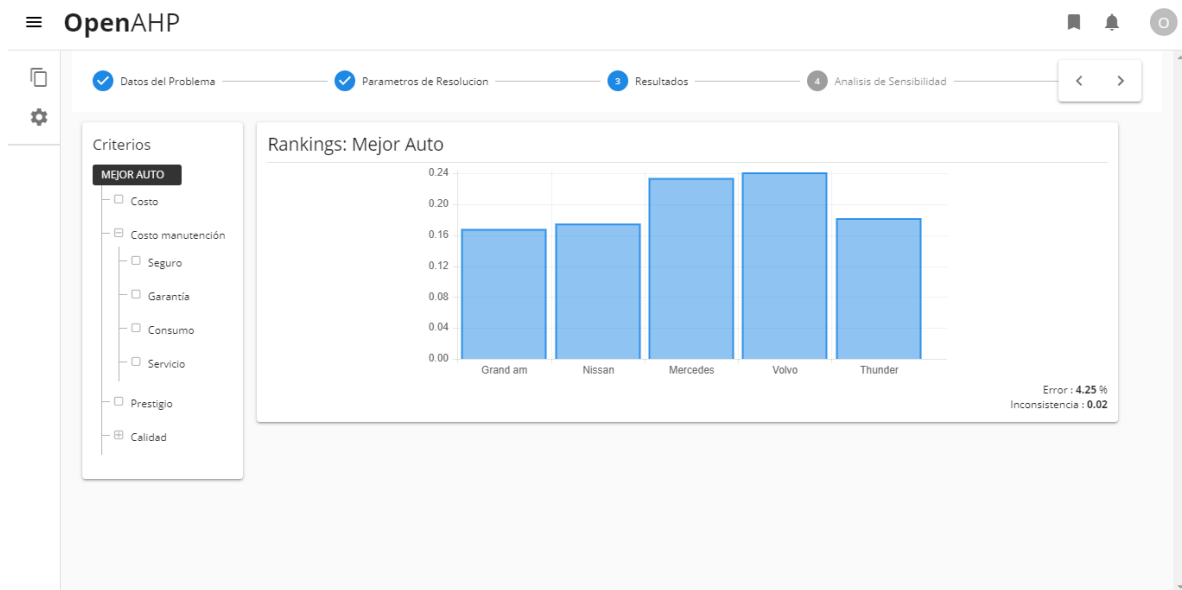


Ilustración 13: Resultados obtenidos en la plataforma
Fuente: Captura de la plataforma

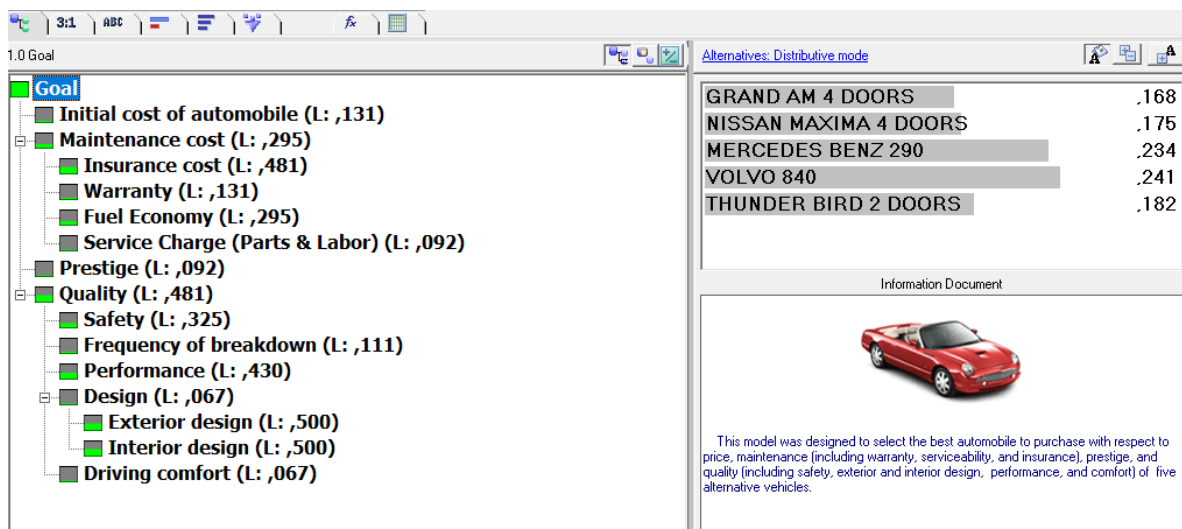


Ilustración 14: Resultados pruebas Expert Choice
Fuente: Captura de Expert Choice

DISEÑO E IMPLEMENTACION DE UNA PLATAFORMA WEB PARA LA RESOLUCION DE PROBLEMAS CON MULTIPLES CRITERIOS

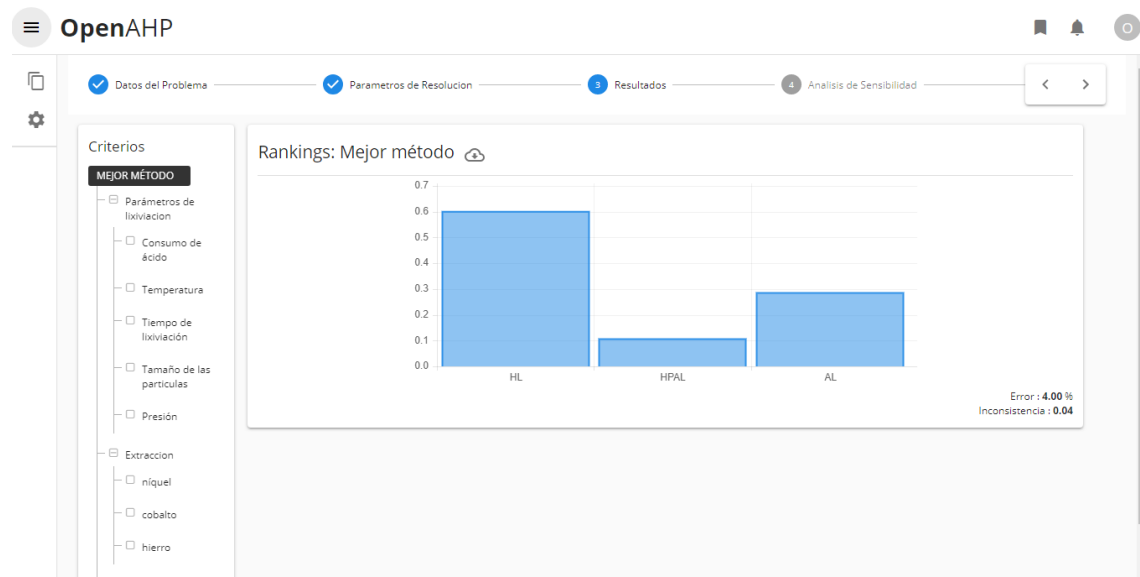


Ilustración 15: Resultados obtenidos en la plataforma
Fuente: Captura de la plataforma

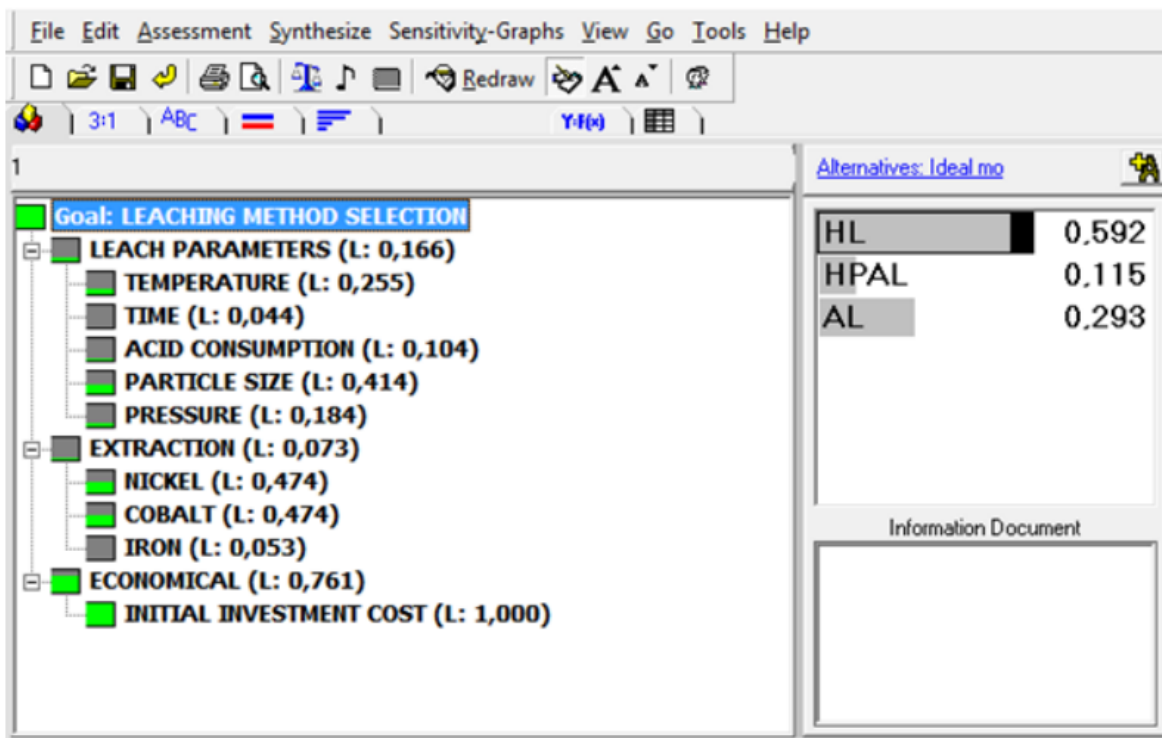


Ilustración 16: Resultados pruebas Expert Choice
Fuente: [16]

4.2 Pruebas de respuesta

Basado en el trabajo realizado en [17] que se basa en el trabajo de Nielsen [18] sobre los tiempos de respuesta, se rescatan los siguientes tiempos a considerar:

- 0.1 segundo: es el límite para que el usuario sienta que el sistema está reaccionando instantáneamente, lo que significa que no es necesario un feedback especial, excepto para mostrar el resultado.
- 1.0 segundo: es el límite para que el flujo de pensamiento del usuario permanezca ininterrumpido, a pesar de que el usuario notará el retraso. Normalmente, no se necesita una retroalimentación especial durante retrasos entre 0.1 y 1.0 segundo, aunque el usuario pierde la sensación de operar directamente sobre los datos.
- 10 segundos es el límite para mantener la atención del usuario centrada en el diálogo. Para demoras mayores, los usuarios querrán realizar otras tareas mientras esperan que la computadora termine.

Usando este análisis se someterá a prueba la plataforma usando la herramienta de desempeño que provee el navegador Google Chrome.

En total para navegar por todos los puntos del problema, es decir acceder a la vista de edición del modelo, la selección de parámetros, resolver el problema, mostrar los resultados y finalmente mostrar el análisis de sensibilidad toma 4.7 segundos y la transición más demorada resulta entre la resolución del problema y mostrar los resultados que toma 0.98 segundos, Con lo que se puede comprobar que el flujo del pensamiento del usuario no será interrumpido y podrá mantener su concentración en todo momento.

CAPÍTULO 5: CONCLUSIONES

Se ha visto que AHP es una poderosa técnica en el ámbito de resolución de problemas de múltiples criterios y pese a los múltiples softwares que se han desarrollado ninguno cumple en totalidad los requerimientos que se han presentado en este mismo trabajo así como en [2] y [13].

Es destacable que previamente han existido variados intentos por construir un software que sea gratuito y que satisfaga a cabalidad los requerimientos. Todas ellas han quedado obsoletas a través del paso del tiempo y algunas son muy específicas en cuanto a el contexto del problema. Con el fin de mitigar estos problemas se ha desarrollado la plataforma de tal manera de que sea fácil de modificar, agregarle o quitarle funcionalidades con el objetivo de incentivar su constante mantención.

5.1 Revisión de objetivos

Como fueron definidos los objetivos de este trabajo en el capítulo 1, se procede a comprobar su satisfacción.

Para los objetivos:

1. Analizar el software existente para DMC:
2. Extraer principales ventajas de cada uno de los softwares para DMC:

Se ha realizado un análisis del software existente exponiendo sus ventajas y falencias (ver capítulo 2.2)

Para los objetivos:

3. Construcción back-end de la plataforma
4. Construcción front-end de la plataforma
6. Integración con API AHP desarrollada en [2]
7. Visualización gráfica de los resultados entregados por la API
8. Generación de reportes con los resultados

Se ha realizado la construcción de una plataforma web modularizada con el fin de que los usuarios de esta puedan mantenerla e integrarle más funcionalidades en un futuro. La integración con la API AHP es completa de manera que los usuarios pueden resolver los problemas y ver los resultados de una manera gráfica y descargar un reporte de ellos

Finalmente, para el objetivo:

5. El software por construir debe ser open-source

Se ha publicado en <https://github.com/o-duarte/OpenAHP> todo el código necesario para la libre ejecución de la plataforma bajo la licencia GNU v3.0

5.2 Trabajo futuro

Queda a completa libertad del usuario de la plataforma desarrollar las mejoras que ellos estimen convenientes, pero se detalla una serie de puntos en que se podría mejorar la plataforma:

1. Existen problemas de decisiones para múltiples tomadores de decisiones [19], estos casos sí son soportados por AHP por lo que es factible realizar la incorporación de la funcionalidad.
2. Actualmente la plataforma provee la exportación de resultados, sería deseable la importación y exportación del modelo que genera estos resultados.
3. Incorporación de más métodos de priorización, cálculo de error, cálculo de consistencia y análisis de sensibilidad, como el índice de consistencia armónico, índice de ambigüedad, comparaciones indirectas, por nombrar algunos.
4. Traducir la plataforma a otros idiomas.

REFERENCIAS

- [1] T. L. Saaty, «How to make a decision: the analytic hierarchy process,» *European Journal of Operational Research*, vol. 48, nº 1, pp. 9-26, 1990.
- [2] D. Quinteros, «DISEÑO E IMPLEMENTACIÓN DE UNA API AHP PARA LA TOMA DE DECISIONES CON MÚLTIPLES CRITERIOS,» UTFSM, 2018.
- [3] B. Vitoriano, «Teoría de la decisión: decisión con incertidumbre, decisión multicriterio y teoría de juegos,» de *Universidad Complutense de Madrid*, Madrid, 2007, pp. 19-41.
- [4] T. L. Saaty, «Decision making with the analytic hierarchy process,» *International Journal of Services Sciences*, vol. 1, pp. 83-98, 2008.
- [5] M. Brunelli, «Introduction to the analytic hierarchy process,» Springer, 2004.
- [6] F. Jiri y K. Aleš, «Judgment Scales and Consistency Measure in AHP,» *Procedia Economics and Finance*, vol. 12, p. 164–173, 2014.
- [7] T. Hurtado y G. Bruno, «El proceso de análisis jerárquico (AHP) como herramienta para la toma de decisiones en la seleccion de proveedores.,» UNMSM, Lima, 2005.
- [8] A. Ishizaka y A. Labib, «Analytic hierarchy process and expert choice: Benefits and limitations.,» *Or Insight*, vol. 7, nº 5.
- [9] R. M. Yunus, «Expert Choice for Ranking Heritage Streets.,» *Procedia - Social and Behavioral Sciences*, vol. 101, pp. 465-475, 2013.
- [10] W. Edwards, «How to use multiattribute utility measurement for social decision making,» *IEEE Transactions on Systems, Man, and Cybernetics*, 1977.
- [11] B. C. D. Opydo, «Makeitrational tutorials,» [En línea]. Available: <http://makeitrational.com/tutorials,>. [Último acceso: 10 junio 2018].
- [12] S. Siraj, L. Mikhailov y J. A. Keane, «Priest: an interactive decision support to estimate priorities from pairwise comparison judgments,» *International Transactions in Operational Research*, vol. 22, nº 2, pp. 217-235, 2015.
- [13] J. Mustajoki y M. Marttunen, «Comparison of Multi-Criteria Decision Analytical Software,» Finnish Environment Institute, 2013.

- [14] NetCraft, «December 2018 Web Server Survey,» [En línea]. Available: <https://news.netcraft.com/archives/2018/12/17/december-2018-web-server-survey.html>. [Último acceso: 31 12 2018].
- [15] K. NOYES, «Docker: A 'Shipping Container' for Linux Code,» [En línea]. Available: <https://www.linux.com/news/docker-shipping-container-linux-code>. [Último acceso: 31 12 2018].
- [16] S. Kursunoglu, Z. T. Ichlas y M. Kaya, «Leaching method selection for Caldag lateritic nickel ore by the analytic hierarchy process,» *Hydrometallurgy*, vol. 171, pp. 179-184, 2017.
- [17] F. F.-H. Nah, «A study on tolerable waiting time: how long are Web users willing to wait?,» Lincoln, 2004.
- [18] J. Nielsen, «Response times: the three important limits.,» de *Usability Engineering*, 1993, p. Capítulo 5.
- [19] T. L. Saaty., «Group decision making and the ahp. In The analytic hierarchy process,» *Computers & Operations Research*, vol. 23, nº 1, pp. 27-35, 1989.
- [20] R. Leonelli, «Enhancing a decision support tool with sensitivity analysis,» University of Manchester, UK, 2012.
- [21] S. Siraj, «Preference elicitation from pairwise comparisons in multi-criteria decision making,» University of Manchester, 2012.

ANEXOS